THE AMAZING BI-PLOT

LAF "WRX2"
Contains files
"WRX21"
"WRX22"

WRX21
PG. 1

We all know, of course, that the daffy-nition of BI-PLOT is "the purchasing of a devious plan." Or any number of other bizarre things, which I'll leave you to decipher.

Ah, but let's get back to the real world. This article gives a demo of a way to do high-resolution grey-scale graphics. That's right! Though I may be stretching the definition, we'll actually see the ZX81/TS1000 display a high-res scale of three shades - black, white and grey. A quote I just invented runs, "Three shades doth verily a grey-scale maketh."

We'll once again use bizarre methods to put that remarkable ZX hardware/software combination to work, deviously planning to make it do things even that IT never thought possible. Best of all, you won't have to purchase a thing (assuming that you're with us so far on this high-res stuff).

For the sake of the demo, let's go back in time. Way back.... do you remember SWN Vol. 1, kids? This contained a lot of (sometimes embarrassing) "early Fred Nachbaur," including a "GreyPlot demo" that I also used in my CE-AMP program (now in shareware). Well, what could be better than to repeat that early demo, using this modification of WRX16?

WHAT IT DOES

BIPLOT works much like "QUICK" mode (SWN 5:2) does, in the sense that it uses the built-in FRAMES counter (FRMS in Hot-Z-ese) to display different things in alternate fields of the TV display. Unlike QUICK, though, we have a continuous display, even though the speed is no faster than normal SLOW mode. (The good news is, we can use FAST mode to do all our plotting, then get into compute-and-display mode to view and play with the result.)

The way we do it, is to toggle between two different high-res display files. Anything plotted in both display files shows up as black (or white in inverse mode). Anything plotted in only one display file will only be displayed every other time. As a result, these points will look "grey." Anything else is in the background color (white or black, in normal and inverse modes respectively).

In the demo, two sets of axes are drawn, one in each display file. However, different horizontal axes are printed just for demo purposes. Two different functions are plotted into each of the display files; their product is printed in both.

After the display has been created, you can view either hi-res display file, or both. You can also reverse the video at any time.

THE DEMO

This demo displays several of WRX16's unique features:

1: Bit-mapped video file. This makes our plotting, etc. extremely compact (the whole mini-operating system takes less than 256 bytes). It's memory-mapped, allowing us write to it from BASIC, using POKE.

2: Any or all vertical columns can be reversed. To reverse a column, change the corresponding element of the dummy display file to 128 (decimal). Our "reverse" (RVRS) routine simply toggles all 32 locations (columns) between 0 and 128 whenever it is called.

3: WRx16 is extremely adaptable. We can change the number of hi-res and low-res lines at will. We can do different things in different fields of the TV display. We can even do "QUICK" mode! (Topic for a future installment, if the interest is there.)

4: Since a full hi-res bit map of a 256*192 screen takes 6K, and we only have 8K available, we have to cut down the size of the hi-res portion to 2/3 of its full size. This gives two 4K blocks that can be used for two display files. With the high-res display files in high memory (as with Tim Stoddard's setup or with Delta), these could possibly be brought back to full size. However, this version demonstrates yet another neat trait of WRx16: you can display any combination of hi-res/low-res rows on one screen.

ENTERING THE PROGRAM

THE EASY WAY

One way to do this one is from the ground up. However, if you have followed the demos so far, you can save some time. Load the demo program of 4:2, and delete the demo portion (everything between and including lines 7 through 540). Leave line 5 in place. Enter line 4,

```
4 REM BIPLOT
```

Perform the following POKEs:

```
POKE 16645,4
POKE 16646,1
POKE 16419,4
```

Then delete line 5. This will "melt together" the REM lines 2 and 3, giving us enough room for the modified service routines. It also keeps us from getting stuck in the long REM lines at the beginning. Henceforth, you may replace line 4, but not delete it. If you do you'll have to pull the plug to regain control. If you ever get in trouble after LIST, and the listing is stuck, LIST 4 and then POKE 16419,4. That'll fix it.

WRx21
PG_4

FROM THE GROUND UP

If you start from scratch, enter a 1 REM line exactly 128 characters long. This is four full lines on the screen, looking like this:

```
  1 REM xxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxx
```

Enter another REM line, numbered 2, long enough to hold at least our present operating system. To leave some elbow-room for any of your additions later, make it, say, 512 bytes long. (FAST mode will make this somewhat quicker.) Then enter the loader program of Listing 2. RUN this to input the values of Table 2 into the 1 REM, and the values from table 3 into the 2 REM.

THE BASIC

Now enter the BASIC program of listing 4. After saving with RUN 9990, the program will auto-run.

Have fun with this. As a final note, let us know if you want to see more reports of new developments. Some possible topics that already exist include: Using SHR-Extended BASIC from machine-code, to end up with a smooth, believable hi-res implementation of the classic SPACE INVADERS game. Using "QUICK" mode with WRx16, or with quasi-hi-res programs. PRINTing characters in high-res.

So once again it's up to you; what do you want us to share with you?

THEORY

Listing 3 shows the disassembly of the modified core and service routines. To make the dual display work, we first have to make some changes to the WRx16 core. From 40BAh to 40C5, we have added a new routine which checks bit 0 of FRAMES. The subsequent commands cause HL to be loaded with 2000h (HRDF #1) or 3000h (HRDF #2), depending on the state of this bit.

When viewing only one display file, the command at 40BD is changed from AND 01 to either LD A,00 or LD A,01. These commands take the same time to execute, so this was just an easy way to insure that the loop will run the same regardless of whether we've viewing just one, or both of the display files.

Note that the command at 40AD was changed, and a NOP was added at 40C5 to compensate for the extra time it takes to run our new code. If you count T-states, you'll see that the overall timing remains exactly the same.

Note also that we load B with only 80h at 40B2, since we are only displaying 128 hi-res lines.

The only other changes are at 40E3 and 40ED. These determine how many low-res rows are displayed. Since the hi-res portion takes 16 rows, we can display 8 lo-res rows for the same 24-row overall display size. However, one row is lost in returning from our hi-res system; this is why we have to kill some time at the start of DP-3, to wait for the start of the first line in the next available row.

The value loaded into DE at 40E3 represents 8000h plus the offset from the start of the normal display file, to the first byte of the starting low-res row to be displayed. The command at 40ED loads B with one more than the number of low-res rows, and C with 7 (always).

Just as our service routine modifies the command at 40BD to change viewing modes, the commands at 40E3 and 40ED could be changed to give a low-res "windowing" option, allowing you to scroll the entire low-res display up and down in the 7-row low-res portion of the display. Use a SYNC routine similar to ours here, to insure that both changes are made right after a vertical sync pulse, since you want to be sure that both changes will be in place by the start of the next field.

TEXT FILE
WRX22
PG. 1

TABLE 2: BI-PLOT WRX16 CORE

START=16514 BYTES=128

```
237:079:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:221:233:243:062:007:071:
000:016:253:198:242:060:032:253:
006:128:017:032:000:033:000:032:
058:052:064:230:001:023:023:023:
023:132:103:000:221:033:204:064:
024:008:017:032:000:005:202:219:
064:025:124:237:071:125:195:130:
192:237:082:237:082:035:042:012:
064:017:049:130:025:062:030:237:
071:062:245:001:007:008:205:181:
002:205:146:002:205:032:002:221:
033:166:064:195:164:002:000:000:
```

TABLES 2 & 3

LOAD "WRX-T23"

TABLE 3: BIPLOT SERVICE ROUTINES

START=16648 BYTES=177

```
033:062:000:024:008:033:062:001:
024:003:033:230:001:034:189:064:
205:168:065:221:033:166:064:201:
205:168:065:221:033:129:002:201:
000:033:000:032:024:003:033:000:
048:001:000:016:054:000:011:084:
093:019:237:176:201:033:132:064:
001:032:000:126:198:128:119:024:
237:038:032:024:022:038:048:024:
018:205:073:065:024:247:038:032:
024:015:038:048:024:011:205:086:
065:024:247:253:203:059:214:024:
004:253:203:059:150:237:075:054:
064:062:128:061:144:218:173:014:
087:089:203:058:203:027:203:058:
203:027:203:058:203:027:046:000:
025:086:121:230:007:060:001:000:
008:203:002:203:017:061:032:012:
253:203:059:086:040:004:203:193:
024:002:203:129:016:235:113:201:
205:043:015:253:070:052:058:052:
064:184:040:250:062:030:237:071:
201:
```

```
40A6~F3       DPLY DI
40A7 3E07     TILO LD A,07
40A9 47       TILO LD B,A
40AA 00       DP-0 NOP
40AB 10FD          DJNZ DP-0
40AD C6F2     TILO ADD A,F2
40AF 3C       DP-A INC A
40B0 20FD          JR NZ DP-A
40B2 0680          LD B,80
40B4 112000   DELY LD DE,0020
40B7 210020        LD HL,HRD1
40BA 3A3440        LD A,(FRMS)
40BD E601     BIT0 AND 01
40BF 17            RLA
40C0 17            RLA
40C1 17            RLA
40C2 17       M*16 RLA
40C3 84       1OR2 ADD A,H
40C4 67            LD H,A
40C5 00       DELY NOP
40C6 DD21CC40      LD IX,DP-1
40CA 1808          JR DP-2
40CC 112000   DP-1 LD DE,0020
40CF 05            DEC B
40D0 CADB40        JP Z DP-3
40D3 19            ADD HL,DE
40D4 7C       DP-2 LD A,H
40D5 ED47          LD I,A
40D7 7D            LD A,L
40D8 C382C0        JP C082
40DB ED52     DP-3 SBC HL,DE
40DD ED52          SBC HL,DE
40DF 23            INC HL
40E0 2A0C40        LD HL,(DFIL)
40E3 113182        LD DE,8231
40E6 19            ADD HL,DE
40E7 3E1E          LD A,1E
40E9 ED47          LD I,A
40EB 3EF5          LD A,F5
40ED 010708        LD BC,0807
40F0 CDB502        CALL 02B5
40F3 CD9202        CALL 0292
40F6 CD2002        CALL 0220
40F9 DD21A640      LD IX,DPLY
40FD C3A402        JP POPS
4100 00            NOP
4101 00            NOP
4102 76            HALT
4103 00            NOP
```

LISTING 3

PG. 1

(not on tape)

```
4108~213E00    DPL1 LD HL,003E
410B 1808           JR STOR
410D 213E01    DPL2 LD HL,LDA1
4110 1803           JR STOR
4112 21E601    DP12 LD HL,01E6
4115 22BD40    STOR LD (BIT0),HL
4118 CDA841    HRES CALL 41A8
411B DD21A640       LD IX,DPLY
411F C9             RET
4120 CDA841    NRML CALL 41A8
4123 DD218102       LD IX,0281
4127 C9             RET
4128 00             NOP
4129 210020    CLS1 LD HL,HRD1
412C 1803           JR CLS*
412E 210030    CLS2 LD HL,HRD2
4131 010010    CLS* LD BC,1000
4134 3600           LD (HL),00
4136 0B        FILL DEC BC
4137 54             LD D,H
4138 5D             LD E,L
4139 13             INC DE
413A EDB0           LDIR
413C C9             RET
413D 218440    RVRS LD HL,COL1
4140 012000         LD BC,0020
4143 7E             LD A,(HL)
4144 C680           ADD A,80
4146 77             LD (HL),A
4147 18ED           JR FILL
4149 2620      PLT1 LD H,20
414B 1816           JR PLOT
414D 2630      PLT2 LD H,30
414F 1812           JR PLOT
4151 CD4941    PL12 CALL PLT1
4154 18F7           JR PLT2
4156 2620      UPL1 LD H,20
4158 180F           JR UNPL
415A 2630      UPL2 LD H,30
415C 180B           JR UNPL
415E CD5641    UP12 CALL UPL1
4161 18F7           JR UPL2
4163 FDCB3BD6  PLOT SET 2,(CDFG)
4167 1804           JR PLT*
```

LISTING 3
PG. 2

```
4169~FDCB3B96 UNPL RES 2,(CDFG)
416D ED4B3640 PLT* LD BC,(COOR)
4171 3E80          LD A,80
4173 3D            DEC A
4174 90            SUB B
4175 DAAD0E        JP C ERRB
4178 57       GADR LD D,A
4179 59            LD E,C
417A CB3A          SRL D
417C CB1B          RR E
417E CB3A          SRL D
4180 CB1B          RR E
4182 CB3A          SRL D
4184 CB1B          RR E
4186 2E00          LD L,00
4188 19            ADD HL,DE
4189 56            LD D,(HL)
418A 79            LD A,C
418B E607          AND 07
418D 3C            INC A
418E 010008        LD BC,0800
4191 CB02     BTLP RLC D
4193 CB11          RL C
4195 3D            DEC A
4196 200C          JR NZ 41A4
4198 FDCB3B56      BIT 2,(CDFG)
419C 2804          JR Z RSBT
419E CBC1     STBT SET 0,C
41A0 1802          JR 41A4
41A2 CB81     RSBT RES 0,C
41A4 10EB          DJNZ BTLP
41A6 71            LD (HL),C
41A7 C9            RET
41A8 CD2B0F        CALL SLOW
41AB FD4634        LD B,(FRMS)
41AE 3A3440        LD A,(FRMS)
41B1 B8            CP B
41B2 28FA          JR Z 41AE
41B4 3E1E          LD A,1E
41B6 ED47          LD I,A
41B8 C9            RET
41B9 00            NOP
41BA 00            NOP
41BB 00            NOP
```

LISTING 3
PG. 3

```
  5 REM WRX16-V2D DUAL HIRES
 10 LET HR1 =16648
 11 LET HR2 =16653
 12 LET HR12=16658
 13 LET HRES=16664
 14 LET NRML=16672
 15 LET CLS1=16681
 16 LET CLS2=16686
 17 LET RVRS=16701
 18 LET PLT1=16713
 19 LET PLT2=16717
 20 LET PL12=16721
 21 LET UPL1=16726
 22 LET UPL2=16730
 23 LET UP12=16734
 24 LET XP  =16438
 25 LET YP  =16439
100 REM greyplot demo
105 FAST
108 REM CLEAR BOTH DF"S
110 RAND USR CLS1
120 RAND USR CLS2
130 FOR I=10240 TO 10271
135 REM H-LINE, DF1
140 POKE I,255
145 REM H-LINE,DF2
150 POKE I+5984,255
160 NEXT I
165 REM V-LINE, BOTH DF"S
170 POKE XP,4
180 FOR I=0 TO 127
190 POKE YP,I
200 RAND USR PL12
210 NEXT I
215 REM V-MARKERS
220 FOR I=10080 TO 8192 STEP -192
230 POKE I,60
232 POKE I+2048,60
235 POKE I+4096,60
237 POKE I+6144,60
240 NEXT I
245 REM H-MARKERS
250 FOR I=4 TO 244 STEP 10
260 POKE XP,I
270 FOR J=2 TO 6
280 POKE YP,J
290 RAND USR PLT2
300 POKE YP,J+59
310 RAND USR PLT1
320 NEXT J
330 NEXT I
340 REM LO-RES AXES
350 FOR I=0 TO 63
360 PLOT I,12
370 PLOT I,27
375 IF I>12 AND I<43 THEN PLOT 0,I
380 NEXT I
```

LISTING 4

PG_1

LOAD "BPDEM1"
(big printer)

OR "BPDEMØ"
(TS2040)

```
 390 REM PLOT CURVES
 395 REM F1:SINE CURVE
 400 LET A$=" SIN (H* PI /40)"
 405 REM F2:DECAYING EXPONENTIAL
 410 LET B$=" EXP (-H/80)"
 415 FOR H=0 TO 251
 420 LET Z1= VAL A$
 425 LET Z2= VAL B$
 430 POKE XP,H+4
 435 REM FUNCTION NO. 1 (DF 1)
 440 POKE YP,63+60*Z1
 445 RAND USR PLT1
 450 REM FUNCTION NO.2 (DF 2)
 455 POKE YP,63+60*Z2
 460 RAND USR PLT2
 465 REM FCTN.1 * FCTN.2 (BOTH)
 470 POKE YP,63+60*Z1*Z2
 475 RAND USR PL12
 480 REM LO-RES PLOTTING
 485 PLOT INT (H/4)+1,27+15*Z1
 490 PLOT INT (H/4)+1,27+15*Z2
 495 PLOT INT (H/4)+1,27+15*Z1*Z2
 520 NEXT H
 525 POKE 16418,0
 590 PRINT AT 17,0;
 593 PRINT "********************************"
 594 PRINT ":1: DISPLAY DF1  2: DISPLAY DF2:"
 595 PRINT ":3: DISPLAY BOTH DISPLAY FILES :"
 596 PRINT ":4: REVERSE HIGH - RES DISPLAYS:"
 597 PRINT ":5: RETURN TO NORMAL LR DISPLAY:"
 598 PRINT ":6: QUIT TO BASIC"; TAB 31;":"
 599 PRINT "********************************"
 600 RAND USR HR12
 610 IF INKEY$ <> "" THEN GOTO 610
 620 LET Z$= INKEY$
 630 IF Z$="" THEN GOTO 620
 640 IF Z$="1" THEN RAND USR HR1
 650 IF Z$="2" THEN RAND USR HR2
 660 IF Z$="3" THEN RAND USR HR12
 670 IF Z$="4" THEN RAND USR RVRS
 680 IF Z$="5" THEN RAND USR NRML
 690 IF Z$ <> "6" THEN GOTO 610
 700 RAND USR NRML
 710 STOP
9990 LET S$="BIPLOT-DEMO"
9991 SAVE S$
9992 PRINT S$
9993 PRINT ,,"BE SURE THAT 8-16K SRAM ENABLED,
HEN PRESS ANY KEY TO START DEMO"
9994 PAUSE 4E4
9995 CLS
9996 PRINT "PLEASE WAIT ABOUT 75 SECONDS...."
9997 PAUSE 200
9998 CLS
9999 RUN
```

LISTING 4

Pg. 2

```
40A6~F3       DPLY DI
40A7 3E07     TILO LD A,07
40A9 47       TILO LD B,A
40AA 00       DP-0 NOP
40AB 10FD          DJNZ DP-0
40AD C6F2     TILO ADD A,F2
40AF 3C       DP-A INC A
40B0 20FD          JR NZ DP-A
40B2 0680          LD B,80
40B4 112000   DELY LD DE,0020
40B7 210020        LD HL,HRD1
40BA 3A3440        LD A,(FRMS)
40BD E601     BIT0 AND 01
40BF 17            RLA
40C0 17            RLA
40C1 17            RLA
40C2 17       M*16 RLA
40C3 84       10R2 ADD A,H
40C4 67            LD H,A
40C5 00       DELY NOP
40C6 DD21CC40      LD IX,DP-1
40CA 1808          JR DP-2
40CC 112000   DP-1 LD DE,0020
40CF 05            DEC B
40D0 CADB40        JP Z DP-3
40D3 19            ADD HL,DE
40D4 7C       DP-2 LD A,H
40D5 ED47          LD I,A
40D7 7D            LD A,L
40D8 C382C0        JP C082
40DB ED52     DP-3 SBC HL,DE
40DD ED52          SBC HL,DE
40DF 23            INC HL
40E0 2A0C40        LD HL,(DFIL)
40E3 113182        LD DE,8231
40E6 19            ADD HL,DE
40E7 3E1E          LD A,1E
40E9 ED47          LD I,A
40EB 3EF5          LD A,F5
40ED 010708        LD BC,0807
40F0 CDB502        CALL 02B5
40F3 CD9202        CALL 0292
40F6 CD2002        CALL 0220
40F9 DD21A640      LD IX,DPLY
40FD C3A402        JP POPS

4100~213E00   DPL1 LD HL,003E
4103 1808          JR STOR
4105 213E01   DPL2 LD HL,LDA1
4108 1803          JR STOR
410A 21E601   DP12 LD HL,01E6
410D 22BD40   STOR LD (BIT0),HL
4110 CDA841   HRES CALL 41A8
4113 DD21A640      LD IX,DPLY
4117 C9            RET
4118 CDA841   NRML CALL 41A8
411B DD218102      LD IX,0281
411F C9            RET
4120 00            NOP
4121 210020   CLS1 LD HL,HRD1
4124 1803          JR CLS*
4126 210030   CLS2 LD HL,HRD2
4129 010010   CLS* LD BC,1000
412C 3600          LD (HL),00
412E 0B       FILL DEC BC
412F 54            LD D,H
4130 5D            LD E,L
4131 13            INC DE
4132 EDB0          LDIR
4134 C9            RET
4135 218440   RVRS LD HL,COL1
4138 012000        LD BC,0020
413B 7E            LD A,(HL)
413C C680          ADD A,80
413E 77            LD (HL),A
413F 18ED          JR FILL
4141 2620     PLT1 LD H,20
4143 1816          JR PLOT
4145 2630     PLT2 LD H,30
4147 1812          JR PLOT
4149 CD4941   PL12 CALL PLT1
414C 18F7          JR PLT2
414E 2620     UPL1 LD H,20
4150 180F          JR UNPL
4152 2630     UPL2 LD H,30
4154 180B          JR UNPL
4156 CD5641   UP12 CALL UPL1
4159 18F7          JR UPL2
415B FDCB3BD6 PLOT SET 2,(CDFG)
415F 1804          JR PLT*
4161~FDCB3B96 UNPL RES 2,(CDFG)
4165 ED4B3640 PLT* LD BC,(COOR)
4169 3E80          LD A,80
416B 3D            DEC A
416C 90            SUB B
416D DAAD0E        JP C ERRB
4170 57       GADR LD D,A
4171 59            LD E,C
4172 CB3A          SRL D
4174 CB1B          RR E
4176 CB3A          SRL D
4178 CB1B          RR E
417A CB3A          SRL D
417C CB1B          RR E
417E 2E00          LD L,00
4180 19            ADD HL,DE
4181 56            LD D,(HL)
4182 79            LD A,C
4183 E607          AND 07
4185 3C            INC A
4186 010008        LD BC,0800
4189 CB02     BTLP RLC D
418B CB11          RL C
418D 3D            DEC A
418E 200C          JR NZ 41A4
4190 FDCB3B56      BIT 2,(CDFG)
4194 2804          JR Z RSBT
4196 CBC1     STBT SET 0,C
4198 1802          JR 41A4
419A CB81     RSBT RES 0,C
419C 10EB          DJNZ BTLP
419E 71            LD (HL),C
419F C9            RET
41A0 CD2B0F        CALL SLOW
41A3 FD4634        LD B,(FRMS)
41A6 3A3440        LD A,(FRMS)
41A9 B8            CP B
41AA 28FA          JR Z 41A6
41AC 3E1E          LD A,1E
41AE ED47          LD I,A
41B0 C9            RET
```

```
  5 REM WRX16-V2D DUAL HIRES
 10 LET HR1 =16648
 11 LET HR2 =16653
 12 LET HR12=16658
 13 LET HRES=16664
 14 LET NRML=16672
 15 LET CLS1=16681
 16 LET CLS2=16686
 17 LET RVRS=16701
 18 LET PLT1=16713
 19 LET PLT2=16717
 20 LET PL12=16721
 21 LET UPL1=16726
 22 LET UPL2=16730
 23 LET UP12=16734
 24 LET XP  =16438
 25 LET YP  =16439
100 REM greyplot demo
105 FAST
108 REM CLEAR BOTH DF"S
110 RAND USR CLS1
120 RAND USR CLS2
130 FOR I=10240 TO 10271
135 REM H-LINE, DF1
140 POKE I,255
145 REM H-LINE,DF2
150 POKE I+5984,255
160 NEXT I
165 REM V-LINE, BOTH DF"S
170 POKE XP,4
180 FOR I=0 TO 127
190 POKE YP,I
200 RAND USR PL12
210 NEXT I
215 REM V-MARKERS
220 FOR I=10080 TO 8192 STEP -192
230 POKE I,60
232 POKE I+2048,60
235 POKE I+4096,60
237 POKE I+6144,60
240 NEXT I
245 REM H-MARKERS
250 FOR I=4 TO 244 STEP 10
260 POKE XP,I
270 FOR J=2 TO 6
280 POKE YP,J
290 RAND USR PLT2
300 POKE YP,J+59
310 RAND USR PLT1
320 NEXT J
330 NEXT I
340 REM LO-RES AXES
350 FOR I=0 TO 63
360 PLOT I,12
370 PLOT I,27
375 IF I)12 AND I(43 THEN PLOT 0,I
380 NEXT I
390 REM PLOT CURVES
395 REM F1:SINE CURVE
400 LET A$=" SIN (H* PI /40)"
405 REM F2:DECAYING EXPONENTIAL
410 LET B$=" EXP (-H/80)"
415 FOR H=0 TO 251
420 LET Z1= VAL A$
425 LET Z2= VAL B$
430 POKE XP,H+4
435 REM FUNCTION NO. 1 (DF 1)
440 POKE YP,63+60*Z1
445 RAND USR PLT1
450 REM FUNCTION NO.2 (DF 2)
455 POKE YP,63+60*Z2
460 RAND USR PLT2
465 REM FCTN.1 * FCTN.2 (BOTH)
470 POKE YP,63+60*Z1*Z2
475 RAND USR PL12
480 REM LO-RES PLOTTING
485 PLOT INT (H/4)+1,27+15*Z1
490 PLOT INT (H/4)+1,27+15*Z2
495 PLOT INT (H/4)+1,27+15*Z1*Z2
520 NEXT H
525 POKE 16418,0
590 PRINT AT 17,0;
593 PRINT "********************************"
594 PRINT ":1: DISPLAY DF1  2: DISPLAY DF2:"
595 PRINT ":3: DISPLAY BOTH DISPLAY FILES :"
596 PRINT ":4: REVERSE HIGH - RES DISPLAYS:"
597 PRINT ":5: RETURN TO NORMAL LR DISPLAY:"
598 PRINT ":6: QUIT TO BASIC"; TAB 31;":"
599 PRINT "********************************"
600 RAND USR HR12
610 IF INKEY$ <> "" THEN GOTO 610
620 LET Z$= INKEY$
630 IF Z$="" THEN GOTO 620
640 IF Z$="1" THEN RAND USR HR1
650 IF Z$="2" THEN RAND USR HR2
660 IF Z$="3" THEN RAND USR HR12
670 IF Z$="4" THEN RAND USR RVRS
680 IF Z$="5" THEN RAND USR NRML
690 IF Z$ <> "6" THEN GOTO 610
700 RAND USR NRML
710 STOP
9990 LET S$="BIPLOT-DEMO"
9991 SAVE S$
9996 PRINT "PLEASE WAIT ABOUT 75 SECONDS...."
9997 PAUSE 200
9998 CLS
9999 RUN
```

LISTING 1:
WRX16 CORE V2.1, CODE DISASSEMBLY

DUMMY DISPLAY FILE

```
addr HEXCODE  NAME MNEMONIC
=========================
4082 ED4F     DUMY LD R,A
4084 00            NOP
4085 00            NOP
4086 00            NOP
4087 00            NOP
4088 00            NOP
4089 00            NOP
408A 00            NOP
408B 00            NOP
408C 00            NOP
408D 00            NOP
408E 00            NOP
408F 00            NOP
4090 00            NOP
4091 00            NOP
4092 00            NOP
4093 00            NOP
4094 00            NOP
4095 00            NOP
4096 00            NOP
4097 00            NOP
4098 00            NOP
4099~00            NOP
409A 00            NOP
409B 00            NOP
409C 00            NOP
409D 00            NOP
409E 00            NOP
409F 00            NOP
40A0 00            NOP
40A1 00            NOP
40A2 00            NOP
40A3 00            NOP
40A4 DDE9          JP (IX)
```

DRIVER CODE

```
40A6~F3       DPLY DI
40A7 3E07     TILO LD A,07
40A9 47       TILO LD B,A
40AA 00       DP-0 NOP
40AB 10FD          DJNZ DP-0
40AD C6EF     TILO ADD A,EF
40AF 3C       DP-A INC A
40B0 20FD          JR NZ DP-A
40B2 06B0          LD B,B0
40B4 112000   DELY LD DE,0020
40B7 210020        LD HL,HRDF
40BA DD21C040      LD IX,DP-1
40BE 1808          JR DP-2
40C0 112000   DP-1 LD DE,0020
40C3 05            DEC B
40C4 CACF40        JP Z DP-3
40C7 19            ADD HL,DE
40C8 7C       DP-2 LD A,H
40C9 ED47          LD I,A
40CB 7D            LD A,L
40CC C382C0        JP C082
40CF ED52     DP-3 SBC HL,DE
40D1 ED52     DELY SBC HL,DE
40D3 23       DELY INC HL
40D4 2A0C40        LD HL,(DFIL)
40D7 11F782        LD DE,82F7
40DA 19            ADD HL,DE
40DB 3E1E          LD A,1E
40DD ED47          LD I,A
40DF 3EF5          LD A,F5
40E1 010702        LD BC,0207
40E4 CDB502        CALL DP-5
40E7 CD9202        CALL MRGN
40EA CD2002        CALL DEND
40ED DD21A640      LD IX,DPLY
40F1 C3A402        JP POPS
40F4 DD21A640 HRES LD IX,DPLY
40F8 C9            RET
40F9 3E1E     NRML LD A,1E
40FB ED47          LD I,A
40FD DD218102      LD IX,0281
4101 C9            RET
```

LISTING 2: MACHINE-CODE LOADER

```
9000 REM LOADER
9010 FAST
9020 CLS
9030 PRINT "START? ";
9040 INPUT ST
9050 PRINT ST,"BYTES? ";
9060 INPUT BY
9070 PRINT BY
9080 FOR N=ST TO ST+BY-1
9090 INPUT V
9100 POKE N,V
9110 PRINT ( STR$ (V+1000))(2 TO );":";
9120 NEXT N
```

TABLE 1: WRX16-V2 CORE (DECIMAL)

START=16514 BYTES=128

```
237:079:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:221:233:243:062:007:071:
000:016:253:198:239:060:032:253:
006:176:017:032:000:033:000:032:
221:033:192:064:024:008:017:032:
000:005:202:207:064:025:124:237:
071:125:195:130:192:237:082:237:
082:035:042:012:064:017:247:130:
025:062:030:237:071:062:245:001:
007:002:205:181:002:205:146:002:
205:032:002:221:033:166:064:195:
164:002:221:033:166:064:201:062:
030:237:071:221:033:129:002:201:
```

OPTIONAL SYNCHRONIZING ROUTINE
CALL THIS BEFORE CHANGING MODES
TO PREVENT SCREEN FLICKER.

```
E.G. CALL SYNC
     JP HRES
OR
     CALL SYNC
     JP NRML
```

```
addr HEXCODE  NAME MNEMONIC
=========================
4105 CD2B0F   SYNC CALL SLOW
4108 FD4634        LD B,(FRMS)
410B 3A3440   NFM? LD A,(FRMS)
410E B8            CP B
410F 28FA          JR Z NFM?
4111 C9            RET
```

TABLE 2: BI-PLOT WRX16 CORE

START=16514 BYTES=128

```
237:079:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:221:233:243:062:007:071:
000:016:253:198:242:060:032:253:
006:128:017:032:000:033:000:032:
058:052:064:230:001:023:023:023:
023:132:103:000:221:033:204:064:
024:008:017:032:000:005:202:219:
064:025:124:237:071:125:195:130:
192:237:082:237:082:035:042:012:
064:017:049:130:025:062:030:237:
071:062:245:001:007:008:205:181:
002:205:146:002:205:032:002:221:
033:166:064:195:164:002:000:000:
```

TABLE 3: BIPLOT SERVICE ROUTINES

START=16648 BYTES=177

```
033:062:000:024:008:033:062:001:
024:003:033:230:001:034:189:064:
205:168:065:221:033:166:064:201:
205:168:065:221:033:129:002:201:
000:033:000:032:024:003:033:000:
048:001:000:016:054:000:011:084:
093:019:237:176:201:033:132:064:
001:032:000:126:198:128:119:024:
237:038:032:024:022:038:048:024:
018:205:073:065:024:247:038:032:
024:015:038:048:024:011:205:086:
065:024:247:253:203:059:214:024:
004:253:203:059:150:237:075:054:
064:062:128:061:144:218:173:014:
087:089:203:058:203:027:203:058:
203:027:203:058:203:027:046:000:
025:086:121:230:007:060:001:000:
008:203:002:203:017:061:032:012:
253:203:059:086:040:004:203:193:
024:002:203:129:016:235:113:201:
205:043:015:253:070:052:058:052:
064:184:040:250:062:030:237:071:
201:
```

LISTING 1:
WRX16 CORE V2.1, CODE DISASSEMBLY

DUMMY DISPLAY FILE

```
addr HEXCODE  NAME MNEMONIC
==========================
4082 ED4F     DUMY LD R,A
4084 00            NOP
4085 00            NOP
4086 00            NOP
4087 00            NOP
4088 00            NOP
4089 00            NOP
408A 00            NOP
408B 00            NOP
408C 00            NOP
408D 00            NOP
408E 00            NOP
408F 00            NOP
4090 00            NOP
4091 00            NOP
4092 00            NOP
4093 00            NOP
4094 00            NOP
4095 00            NOP
4096 00            NOP
4097 00            NOP
4098 00            NOP
4099~00            NOP
409A 00            NOP
409B 00            NOP
409C 00            NOP
409D 00            NOP
409E 00            NOP
409F 00            NOP
40A0 00            NOP
40A1 00            NOP
40A2 00            NOP
40A3 00            NOP
40A4 DDE9          JP (IX)
```

DRIVER CODE

```
40A6~F3      DPLY DI
40A7 3E07    TILO LD A,07
40A9 47      TILO LD B,A
40AA 00      DP-0 NOP
40AB 10FD         DJNZ DP-0
40AD C6EF    TILO ADD A,EF
40AF 3C      DP-A INC A
40B0 20FD         JR NZ DP-A
40B2 06B0         LD B,B0
40B4 112000  DELY LD DE,0020
40B7 210020       LD HL,HRDF
40BA DD21C040     LD IX,DP-1
40BE 1808         JR DP-2
40C0 112000  DP-1 LD DE,0020
40C3 05           DEC B
40C4 CACF40       JP Z DP-3
40C7 19           ADD HL,DE
40C8 7C      DP-2 LD A,H
40C9 ED47         LD I,A
40CB 7D           LD A,L
40CC C382C0       JP C082
40CF ED52    DP-3 SBC HL,DE
40D1 ED52    DELY SBC HL,DE
40D3 23      DELY INC HL
40D4 2A0C40       LD HL,(DFIL)
40D7 11F782       LD DE,82F7
40DA 19           ADD HL,DE
40DB 3E1E         LD A,1E
40DD ED47         LD I,A
40DF 3EF5         LD A,F5
40E1 010702       LD BC,0207
40E4 CDB502       CALL DP-5
40E7 CD9202       CALL MRGN
40EA CD2002       CALL DEND
40ED DD21A640     LD IX,DPLY
40F1 C3A402       JP POPS
40F4 DD21A640 HRES LD IX,DPLY
40F8 C9           RET
40F9 3E1E    NRML LD A,1E
40FB ED47         LD I,A
40FD DD218102     LD IX,0281
4101 C9           RET
```

LISTING 2: MACHINE-CODE LOADER

```
9000 REM LOADER
9010 FAST
9020 CLS
9030 PRINT "START? ";
9040 INPUT ST
9050 PRINT ST,"BYTES? ";
9060 INPUT BY
9070 PRINT BY
9080 FOR N=ST TO ST+BY-1
9090 INPUT V
9100 POKE N,V
9110 PRINT ( STR$ (V+1000))(2 TO );":";
9120 NEXT N
```

TABLE 1: WRX16-V2 CORE (DECIMAL)

START=16514 BYTES=128

```
237:079:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:221:233:243:062:007:071:
000:016:253:198:239:060:032:253:
006:176:017:032:000:033:000:032:
221:033:192:064:024:008:017:032:
000:005:202:207:064:025:124:237:
071:125:195:130:192:237:082:237:
082:035:042:012:064:017:247:130:
025:062:030:237:071:062:245:001:
007:002:205:181:002:205:146:002:
205:032:002:221:033:166:064:195:
164:002:221:033:166:064:201:062:
030:237:071:221:033:129:002:201:
```

TABLE 2: BI-PLOT WRX16 CORE

START=16514 BYTES=128

```
237:079:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:000:000:000:000:000:000:
000:000:221:233:243:062:007:071:
000:016:253:198:242:060:032:253:
006:128:017:032:000:033:000:032:
058:052:064:230:001:023:023:023:
023:132:103:000:221:033:204:064:
024:008:017:032:000:005:202:219:
064:025:124:237:071:125:195:130:
192:237:082:237:082:035:042:012:
064:017:049:130:025:062:030:237:
071:062:245:001:007:008:205:181:
002:205:146:002:205:032:002:221:
033:166:064:195:164:002:000:000:
```

OPTIONAL SYNCHRONIZING ROUTINE
CALL THIS BEFORE CHANGING MODES
TO PREVENT SCREEN FLICKER.

```
E.G. CALL SYNC
     JP HRES
OR
     CALL SYNC
     JP NRML

addr HEXCODE  NAME MNEMONIC
==========================
4105 CD2B0F   SYNC CALL SLOW
4108 FD4634        LD B,(FRMS)
410B 3A3440   NFM? LD A,(FRMS)
410E B8            CP B
410F 28FA          JR Z NFM?
4111 C9            RET
```

TABLE 3: BIPLOT SERVICE ROUTINES

START=16648 BYTES=177

```
033:062:000:024:008:033:062:001:
024:003:033:230:001:034:189:064:
205:168:065:221:033:166:064:201:
205:168:065:221:033:129:002:201:
000:033:000:032:024:003:033:000:
048:001:000:016:054:000:011:084:
093:019:237:176:201:033:132:064:
001:032:000:126:198:128:119:024:
237:038:032:024:022:038:048:024:
018:205:073:065:024:247:038:032:
024:015:038:048:024:011:205:086:
065:024:247:253:203:059:214:024:
004:253:203:059:150:237:075:054:
064:062:128:061:144:218:173:014:
087:089:203:058:203:027:203:058:
203:027:203:058:203:027:046:000:
025:086:121:230:007:060:001:000:
008:203:002:203:017:061:032:012:
253:203:059:086:040:004:203:193:
024:002:203:129:016:235:113:201:
205:043:015:253:070:052:058:052:
064:184:040:250:062:030:237:071:
201:
```

```
40A6~F3       DPLY DI
40A7 3E07     TILO LD A,07
40A9 47       TILO LD B,A
40AA 00       DP-0 NOP
40AB 10FD          DJNZ DP-0
40AD C6F2     TILO ADD A,F2
40AF 3C       DP-A INC A
40B0 20FD          JR NZ DP-A
40B2 0680          LD B,80
40B4 112000   DELY LD DE,0020
40B7 210020        LD HL,HRD1
40BA 3A3440        LD A,(FRMS)
40BD E601     BIT0 AND 01
40BF 17            RLA
40C0 17            RLA
40C1 17            RLA
40C2 17       M*16 RLA
40C3 84       10R2 ADD A,H
40C4 67            LD H,A
40C5 00       DELY NOP
40C6 DD21CC40      LD IX,DP-1
40CA 1808          JR DP-2
40CC 112000   DP-1 LD DE,0020
40CF 05            DEC B
40D0 CADB40        JP Z DP-3
40D3 19            ADD HL,DE
40D4 7C       DP-2 LD A,H
40D5 ED47          LD I,A
40D7 7D            LD A,L
40D8 C382C0        JP C082
40DB ED52     DP-3 SBC HL,DE
40DD ED52          SBC HL,DE
40DF 23            INC HL
40E0 2A0C40        LD HL,(DFIL)
40E3 113182        LD DE,8231
40E6 19            ADD HL,DE
40E7 3E1E          LD A,1E
40E9 ED47          LD I,A
40EB 3EF5          LD A,F5
40ED 010708        LD BC,0807
40F0 CDB502        CALL 02B5
40F3 CD9202        CALL 0292
40F6 CD2002        CALL 0220
40F9 DD21A640      LD IX,DPLY
40FD C3A402        JP POPS

4108~213E00   DPL1 LD HL,003E
410B 1808          JR STOR
410D 213E01   DPL2 LD HL,LDA1
4110 1803          JR STOR
4112 21E601   DP12 LD HL,01E6
4115 22BD40   STOR LD (BIT0),HL
4118 CDA841   HRES CALL 41A8
411B DD21A640      LD IX,DPLY
411F C9            RET
4120 CDA841   NRML CALL 41A8
4123 DD218102      LD IX,0281
4127 C9            RET
4128 00            NOP
4129 210020   CLS1 LD HL,HRD1
412C 1803          JR CLS*
412E 210030   CLS2 LD HL,HRD2
4131 010010   CLS* LD BC,1000
4134 3600          LD (HL),00
4136 0B       FILL DEC BC
4137 54            LD D,H
4138 5D            LD E,L
4139 13            INC DE
413A EDB0          LDIR
413C C9            RET
413D 218440   RVRS LD HL,COL1
4140 012000        LD BC,0020
4143 7E            LD A,(HL)
4144 C680          ADD A,80
4146 77            LD (HL),A
4147 18ED          JR FILL
4149 2620     PLT1 LD H,20
414B 1816          JR PLOT
414D 2630     PLT2 LD H,30
414F 1812          JR PLOT
4151 CD4941   PL12 CALL PLT1
4154 18F7          JR PLT2
4156 2620     UPL1 LD H,20
4158 180F          JR UNPL
415A 2630     UPL2 LD H,30
415C 180B          JR UNPL
415E CD5641   UP12 CALL UPL1
4161 18F7          JR UPL2
4163 FDCB3BD6 PLOT SET 2,(CDFG)
4167 1804          JR PLT*
4169~FDCB3B96 UNPL RES 2,(CDFG)
416D ED4B3640 PLT* LD BC,(COOR)
4171 3E80          LD A,80
4173 3D            DEC A
4174 90            SUB B
4175 DAAD0E        JP C ERRB
4178 57       GADR LD D,A
4179 59            LD E,C
417A CB3A          SRL D
417C CB1B          RR E
417E CB3A          SRL D
4180 CB1B          RR E
4182 CB3A          SRL D
4184 CB1B          RR E
4186 2E00          LD L,00
4188 19            ADD HL,DE
4189 56            LD D,(HL)
418A 79            LD A,C
418B E607          AND 07
418D 3C            INC A
418E 010008        LD BC,0800
4191 CB02     BTLP RLC D
4193 CB11          RL C
4195 3D            DEC A
4196 200C          JR NZ 41A4
4198 FDCB3B56      BIT 2,(CDFG)
419C 2804          JR Z RSBT
419E CBC1     STBT SET 0,C
41A0 1802          JR 41A4
41A2 CB81     RSBT RES 0,C
41A4 10EB          DJNZ BTLP
41A6 71            LD (HL),C
41A7 C9            RET
41A8 CD2B0F        CALL SLOW
41AB FD4634        LD B,(FRMS)
41AE 3A3440        LD A,(FRMS)
41B1 B8            CP B
41B2 28FA          JR Z 41AE
41B4 3E1E          LD A,1E
41B6 ED47          LD I,A
41B8 C9            RET
```

LISTING 4
BI-PLOT
BASIC

```
  5 REM WRX16-V2D DUAL HIRES
 10 LET HR1 =16648
 11 LET HR2 =16653
 12 LET HR12=16658
 13 LET HRES=16664
 14 LET NRML=16672
 15 LET CLS1=16681
 16 LET CLS2=16686
 17 LET RVRS=16701
 18 LET PLT1=16713
 19 LET PLT2=16717
 20 LET PL12=16721
 21 LET UPL1=16726
 22 LET UPL2=16730
 23 LET UP12=16734
 24 LET XP  =16438
 25 LET YP  =16439
100 REM greyplot demo
105 FAST
108 REM CLEAR BOTH DF"S
110 RAND USR CLS1
120 RAND USR CLS2
130 FOR I=10240 TO 10271
135 REM H-LINE, DF1
140 POKE I,255
145 REM H-LINE,DF2
150 POKE I+5984,255
160 NEXT I
165 REM V-LINE, BOTH DF"S
170 POKE XP,4
180 FOR I=0 TO 127
190 POKE YP,I
200 RAND USR PL12
210 NEXT I
215 REM V-MARKERS
220 FOR I=10080 TO 8192 STEP -192
230 POKE I,60
232 POKE I+2048,60
235 POKE I+4096,60
237 POKE I+6144,60
240 NEXT I
245 REM H-MARKERS
250 FOR I=4 TO 244 STEP 10
260 POKE XP,I
270 FOR J=2 TO 6
280 POKE YP,J
290 RAND USR PLT2
300 POKE YP,J+59
310 RAND USR PLT1
320 NEXT J
330 NEXT I
340 REM LO-RES AXES
350 FOR I=0 TO 63
360 PLOT I,12
370 PLOT I,27
375 IF I)12 AND I(43 THEN PLOT 0,I
380 NEXT I
390 REM PLOT CURVES
395 REM F1:SINE CURVE
400 LET A$=" SIN (H* PI /40)"
405 REM F2:DECAYING EXPONENTIAL
410 LET B$=" EXP (-H/80)"
415 FOR H=0 TO 251
420 LET Z1= VAL A$
425 LET Z2= VAL B$
430 POKE XP,H+4
435 REM FUNCTION NO. 1 (DF 1)
440 POKE YP,63+60*Z1
445 RAND USR PLT1
450 REM FUNCTION NO.2 (DF 2)
455 POKE YP,63+60*Z2
460 RAND USR PLT2
465 REM FCTN.1 * FCTN.2 (BOTH)
470 POKE YP,63+60*Z1*Z2
475 RAND USR PL12
480 REM LO-RES PLOTTING
485 PLOT INT (H/4)+1,27+15*Z1
490 PLOT INT (H/4)+1,27+15*Z2
495 PLOT INT (H/4)+1,27+15*Z1*Z2
520 NEXT H
525 POKE 16418,0
590 PRINT AT 17,0;
593 PRINT "********************************"
594 PRINT ":1: DISPLAY DF1  2: DISPLAY DF2:"
595 PRINT ":3: DISPLAY BOTH DISPLAY FILES :"
596 PRINT ":4: REVERSE HIGH - RES DISPLAYS:"
597 PRINT ":5: RETURN TO NORMAL LR DISPLAY:"
598 PRINT ":6: QUIT TO BASIC"; TAB 31;":"
599 PRINT "********************************"
600 RAND USR HR12
610 IF INKEY$ <> "" THEN GOTO 610
620 LET Z$= INKEY$
630 IF Z$="" THEN GOTO 620
640 IF Z$="1" THEN RAND USR HR1
650 IF Z$="2" THEN RAND USR HR2
660 IF Z$="3" THEN RAND USR HR12
670 IF Z$="4" THEN RAND USR RVRS
680 IF Z$="5" THEN RAND USR NRML
690 IF Z$ <> "6" THEN GOTO 610
700 RAND USR NRML
710 STOP
9990 LET S$="BIPLOT-DEMO"
9991 SAVE S$
9996 PRINT "PLEASE WAIT ABOUT 75 SECONDS...."
9997 PAUSE 200
9998 CLS
9999 RUN
```